Introducción a la Criptografía Moderna

Firma Digitales

Rodrigo Abarzúa[†],

[†] Universidad de Santiago de Chile
rodrigo.abarzua@usach.cl

May 27, 2014

Firma Digital

- Las firmas digitales son una de las herramientas criptográficas más importantes y que están ampliamente utilizado en la actualidad.
- Las aplicaciones de las firmas digitales van desde certificados digitales para el comercio electrónico seguro, a la firma legal de los contratos para asegurar las actualizaciones de software.
- Junto con el establecimiento de clave por canales inseguros (Diffie-Hellman), forman los dos más importantes ejemplo de la utilidad que tiene de la criptografía de clave pública o Asimétrica.
- Las firmas digitales comparten algunas funciones con las firmas manuscritas reales. En particular, proporcionan un método para asegurar que un mensaje es auténtico a un usuario, es decir, que un usuario A fue de hecho quien genero el mensaje.
- Sin embargo, realmente ofrecen mucha más funcionalidad.

Firma Digital

- Las firmas digitales son una de las herramientas criptográficas más importantes y que están ampliamente utilizado en la actualidad.
- Las aplicaciones de las firmas digitales van desde certificados digitales para el comercio electrónico seguro, a la firma legal de los contratos para asegurar las actualizaciones de software.
- Junto con el establecimiento de clave por canales inseguros (Diffie-Hellman), forman los dos más importantes ejemplo de la utilidad que tiene de la criptografía de clave pública o Asimétrica.
- Las firmas digitales comparten algunas funciones con las firmas manuscritas reales. En particular, proporcionan un método para asegurar que un mensaje es auténtico a un usuario, es decir, que un usuario A fue de hecho quien genero el mensaje.
- Sin embargo, realmente ofrecen mucha más funcionalidad.

Firma Digital

- Las firmas digitales son una de las herramientas criptográficas más importantes y que están ampliamente utilizado en la actualidad.
- Las aplicaciones de las firmas digitales van desde certificados digitales para el comercio electrónico seguro, a la firma legal de los contratos para asegurar las actualizaciones de software.
- Junto con el establecimiento de clave por canales inseguros (Diffie-Hellman), forman los dos más importantes ejemplo de la utilidad que tiene de la criptografía de clave pública o Asimétrica.
- Las firmas digitales comparten algunas funciones con las firmas manuscritas reales. En particular, proporcionan un método para asegurar que un mensaje es auténtico a un usuario, es decir, que un usuario A fue de hecho quien genero el mensaje.
- Sin embargo, realmente ofrecen mucha más funcionalidad.

Firma Digital

- Las firmas digitales son una de las herramientas criptográficas más importantes y que están ampliamente utilizado en la actualidad.
- Las aplicaciones de las firmas digitales van desde certificados digitales para el comercio electrónico seguro, a la firma legal de los contratos para asegurar las actualizaciones de software.
- Junto con el establecimiento de clave por canales inseguros (Diffie-Hellman), forman los dos más importantes ejemplo de la utilidad que tiene de la criptografía de clave pública o Asimétrica.
- Las firmas digitales comparten algunas funciones con las firmas manuscritas reales. En particular, proporcionan un método para asegurar que un mensaje es auténtico a un usuario, es decir, que un usuario A fue de hecho quien genero el mensaje.
- Sin embargo, realmente ofrecen mucha más funcionalidad.

Firma Digital

- Las firmas digitales son una de las herramientas criptográficas más importantes y que están ampliamente utilizado en la actualidad.
- Las aplicaciones de las firmas digitales van desde certificados digitales para el comercio electrónico seguro, a la firma legal de los contratos para asegurar las actualizaciones de software.
- Junto con el establecimiento de clave por canales inseguros (Diffie-Hellman), forman los dos más importantes ejemplo de la utilidad que tiene de la criptografía de clave pública o Asimétrica.
- Las firmas digitales comparten algunas funciones con las firmas manuscritas reales. En particular, proporcionan un método para asegurar que un mensaje es auténtico a un usuario, es decir, que un usuario A fue de hecho quien genero el mensaje.
- Sin embargo, realmente ofrecen mucha más funcionalidad.

Principio de la Firma Digital

- La idea básica es que la persona que firma el mensaje x utiliza una clave privada k_{pr} y la parte receptora utiliza la clave pública k_{pub}.
- El principio de un esquema de firma digital se muestra en la siguiente figura.

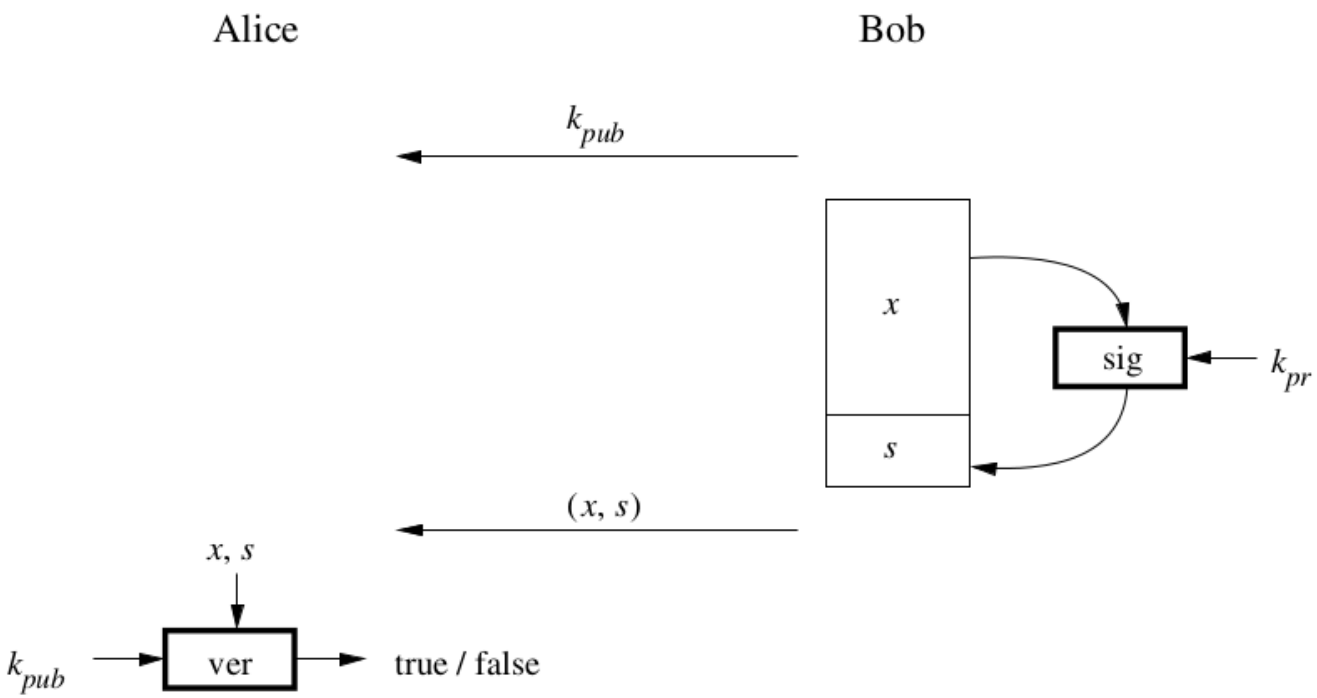


Figure: Principio de la firma digital que implica la firma y verificación de un mensaje

Principio de la Firma Digital

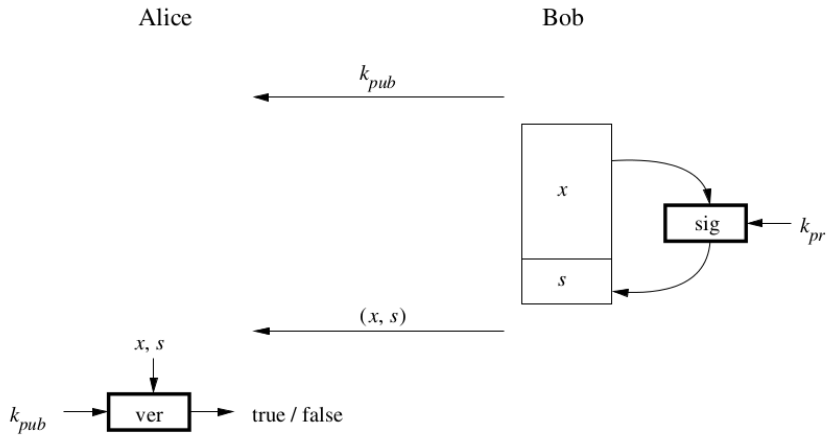


Figure: Principio de la firma digital que implica la firma y verificación de un mensaje

- El proceso se inicia con Bob que firma del mensaje x.
- El algoritmo de firma es una función que depende de la clave privada k_{pr} de Bob y su mensaje x, esto para relacionar el mensaje x y la clave privada k_{pr}.
- Suponiendo que Bob mantiene su clave "privada" k_{pr}, sólo Bob puede firmar un mensaje de x en su nombre.
- Después de firmar el mensaje, la firma s se adjunta al mensaje x y el par (x, s) y es enviado a Alice.

Principio de la Firma Digital

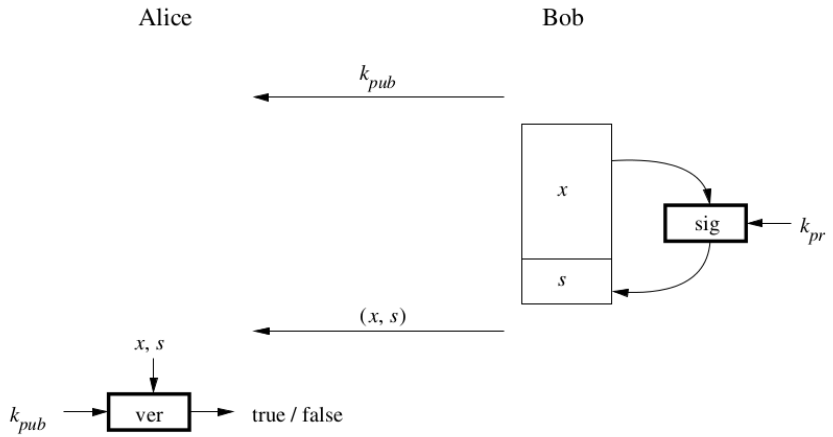


Figure: Principio de la firma digital que implica la firma y verificación de un mensaje

- El proceso se inicia con Bob que firma del mensaje x.
- El algoritmo de firma es una función que depende de la clave privada k_{pr} de Bob y su mensaje x, esto para relacionar el mensaje x y la clave privada k_{pr}.
- Suponiendo que Bob mantiene su clave "privada" k_{pr}, sólo Bob puede firmar un mensaje de x en su nombre.
- Después de firmar el mensaje, la firma s se adjunta al mensaje x y el par (x, s) y es enviado a Alice.

Principio de la Firma Digital

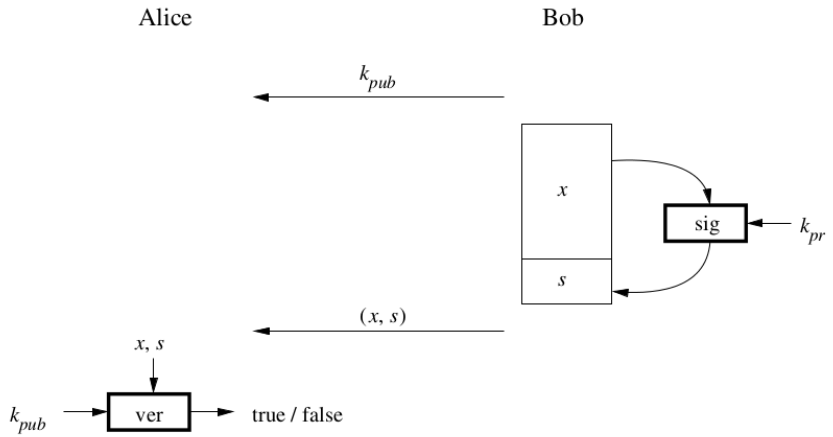


Figure: Principio de la firma digital que implica la firma y verificación de un mensaje

- El proceso se inicia con Bob que firma del mensaje x.
- El algoritmo de firma es una función que depende de la clave privada k_{pr} de Bob y su mensaje x, esto para relacionar el mensaje x y la clave privada k_{pr}.
- Suponiendo que Bob mantiene su clave "privada" k_{pr}, sólo Bob puede firmar un mensaje de x en su nombre.
- Después de firmar el mensaje, la firma s se adjunta al mensaje x y el par (x, s) y es enviado a Alice.

Principio de la Firma Digital

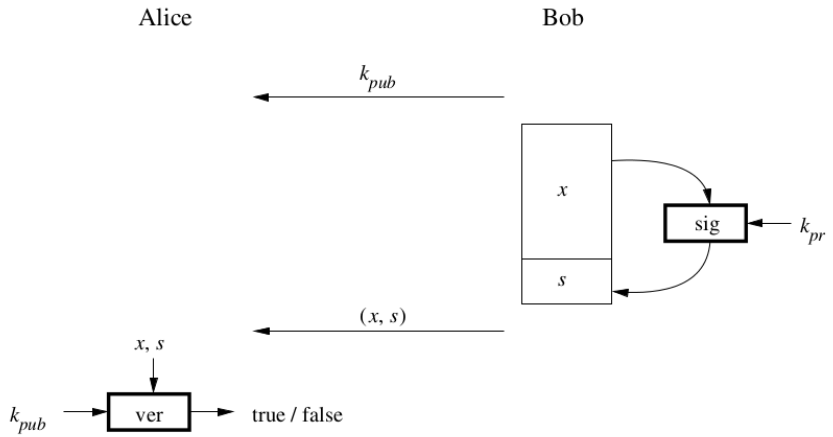


Figure: Principio de la firma digital que implica la firma y verificación de un mensaje

- El proceso se inicia con Bob que firma del mensaje x.
- El algoritmo de firma es una función que depende de la clave privada k_{pr} de Bob y su mensaje x, esto para relacionar el mensaje x y la clave privada k_{pr}.
- Suponiendo que Bob mantiene su clave "privada" k_{pr}, sólo Bob puede firmar un mensaje de x en su nombre.
- Después de firmar el mensaje, la firma s se adjunta al mensaje x y el par (x, s) y es enviado a Alice.

Principio de la Firma Digital

Observación

Es importante tener en cuenta que una firma digital por sí mismo no sirve de nada si no va acompañado por el mensaje. Una firma digital sin el mensaje es el equivalente de la firma manuscrita en una tira de papel sin contrato o un cheque que se supone que debe ser firmado.

- La firma digital es un número entero grande, por ejemplo, una cadena de 2048-bits.
- La firma sólo es útil a Alice, si ella tiene medios para verificar si la firma es válida o no.
- Para esto, se necesita una función de verificación que tiene tanto a x como la firma s como entradas. Con el fin de vincular la firma de Bob, la función también requiere su clave pública k_{pub}.
- A pesar de que la función de verificación tiene entradas grandes, su única salida es el estado binario "verdadero" o "falso".
- Si el mensaje x se firmó en realidad con la clave privada k_{pr} que corresponde a la clave de verificación pública k_{pub}, la salida es verdadera, de lo contrario, la salida es falsa

Principio de la Firma Digital

Observación

Es importante tener en cuenta que una firma digital por sí mismo no sirve de nada si no va acompañado por el mensaje. Una firma digital sin el mensaje es el equivalente de la firma manuscrita en una tira de papel sin contrato o un cheque que se supone que debe ser firmado.

- La firma digital es un número entero grande, por ejemplo, una cadena de 2048-bits.
- La firma sólo es útil a Alice, si ella tiene medios para verificar si la firma es válida o no.
- Para esto, se necesita una función de verificación que tiene tanto a x como la firma s como entradas. Con el fin de vincular la firma de Bob, la función también requiere su clave pública k_{pub}.
- A pesar de que la función de verificación tiene entradas grandes, su única salida es el estado binario "verdadero" o "falso".
- Si el mensaje x se firmó en realidad con la clave privada k_{pr} que corresponde a la clave de verificación pública k_{pub}, la salida es verdadera, de lo contrario, la salida es falsa

Principio de la Firma Digital

Observación

Es importante tener en cuenta que una firma digital por sí mismo no sirve de nada si no va acompañado por el mensaje. Una firma digital sin el mensaje es el equivalente de la firma manuscrita en una tira de papel sin contrato o un cheque que se supone que debe ser firmado.

- La firma digital es un número entero grande, por ejemplo, una cadena de 2048-bits.
- La firma sólo es útil a Alice, si ella tiene medios para verificar si la firma es válida o no.
- Para esto, se necesita una función de verificación que tiene tanto a x como la firma s como entradas. Con el fin de vincular la firma de Bob, la función también requiere su clave pública k_{pub}.
- A pesar de que la función de verificación tiene entradas grandes, su única salida es el estado binario "verdadero" o "falso".
- Si el mensaje x se firmó en realidad con la clave privada k_{pr} que corresponde a la clave de verificación pública k_{pub}, la salida es verdadera, de lo contrario, la salida es falsa

Principio de la Firma Digital

Observación

Es importante tener en cuenta que una firma digital por sí mismo no sirve de nada si no va acompañado por el mensaje. Una firma digital sin el mensaje es el equivalente de la firma manuscrita en una tira de papel sin contrato o un cheque que se supone que debe ser firmado.

- La firma digital es un número entero grande, por ejemplo, una cadena de 2048-bits.
- La firma sólo es útil a Alice, si ella tiene medios para verificar si la firma es válida o no.
- Para esto, se necesita una función de verificación que tiene tanto a x como la firma s como entradas. Con el fin de vincular la firma de Bob, la función también requiere su clave pública k_{pub}.
- A pesar de que la función de verificación tiene entradas grandes, su única salida es el estado binario "verdadero" o "falso".
- Si el mensaje x se firmó en realidad con la clave privada k_{pr} que corresponde a la clave de verificación pública k_{pub}, la salida es verdadera, de lo contrario, la salida es falsa

Principio de la Firma Digital

Observación

Es importante tener en cuenta que una firma digital por sí mismo no sirve de nada si no va acompañado por el mensaje. Una firma digital sin el mensaje es el equivalente de la firma manuscrita en una tira de papel sin contrato o un cheque que se supone que debe ser firmado.

- La firma digital es un número entero grande, por ejemplo, una cadena de 2048-bits.
- La firma sólo es útil a Alice, si ella tiene medios para verificar si la firma es válida o no.
- Para esto, se necesita una función de verificación que tiene tanto a x como la firma s como entradas. Con el fin de vincular la firma de Bob, la función también requiere su clave pública k_{pub}.
- A pesar de que la función de verificación tiene entradas grandes, su única salida es el estado binario "verdadero" o "falso".
- Si el mensaje x se firmó en realidad con la clave privada k_{pr} que corresponde a la clave de verificación pública k_{pub}, la salida es verdadera, de lo contrario, la salida es falsa

Principio de la Firma Digital

A partir de estas observaciones generales, podemos fácilmente desarrollar un protocolo de firma digital genérica:

Basic Digital Signature Protocol

Alice		Bob
		generate $k_{pr,B}$, $k_{pub,B}$
	$\xleftarrow{k_{pub,B}}$	publish public key
		sign message: $s = \text{sig}_{k_{pr}}(x)$
	$\xleftarrow{(x,s)}$	send message + signature
verify signature: $\text{ver}_{k_{pr,B}}(x, s) = \text{true/false}$		

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Principio de la Firma Digital

A partir de esta configuración, la propiedad principal de la firma digital son:

- Un mensaje firmado sin ambigüedades se remonta a su creador ya que una firma válida sólo se puede calcular con la clave privada del firmante único.
- Sólo el firmante tiene la capacidad de generar una firma en su nombre. Por lo tanto, podemos probar que el firmante ha generado realmente el mensaje.
- Esta prueba puede incluso tener significado legal, por ejemplo,:
  - Las Firmas Electrónicas en el Comercio Global y Nacional (ESIGN) en los EE.UU.
  - o en el Signaturgesetz, o ley de firma, en Alemania.
- Se debe observar de que el protocolo básico anterior no proporciona ninguna confidencialidad del mensaje ya que el mensaje se está enviando x en el claro.
- Por supuesto, el mensaje puede ser confidencial al cifrarlo, por ejemplo, con AES o 3DES.
- Cada una de las tres familias populares de algoritmos de clave pública, es decir, factorización de entero, logaritmos discretos y curvas elípticas, nos permite construir las firmas digitales.
- En el resto de esta presentación estudiaremos algunos esquemas de firmas que son de relevancia práctica.

Pero antes recordemos:

¿Cuáles son los posibles objetivos de seguridad que un sistema de seguridad puede poseer?

Confidencialidad: La información se mantiene en secreto para todos, menos para las partes autorizadas (Alice y Bob).

Integridad: Los mensajes no se han modificado en el tránsito.

Autenticación de Mensajes: El remitente de un mensaje es auténtico.

No repudio: El emisor de un mensaje no se puede negar la creación del mensaje.

Pero antes recordemos:

¿Cuáles son los posibles objetivos de seguridad que un sistema de seguridad puede poseer?

Confidencialidad: La información se mantiene en secreto para todos, menos para las partes autorizadas (Alice y Bob).

Integridad: Los mensajes no se han modificado en el tránsito.

Autenticación de Mensajes: El remitente de un mensaje es auténtico.

No repudio: El emisor de un mensaje no se puede negar la creación del mensaje.

Pero antes recordemos:

¿Cuáles son los posibles objetivos de seguridad que un sistema de seguridad puede poseer?

Confidencialidad: La información se mantiene en secreto para todos, menos para las partes autorizadas (Alice y Bob).

Integridad: Los mensajes no se han modificado en el tránsito.

Autenticación de Mensajes: El remitente de un mensaje es auténtico.

No repudio: El emisor de un mensaje no se puede negar la creación del mensaje.

Pero antes recordemos:

¿Cuáles son los posibles objetivos de seguridad que un sistema de seguridad puede poseer?

Confidencialidad: La información se mantiene en secreto para todos, menos para las partes autorizadas (Alice y Bob).

Integridad: Los mensajes no se han modificado en el tránsito.

Autenticación de Mensajes: El remitente de un mensaje es auténtico.

No repudio: El emisor de un mensaje no se puede negar la creación del mensaje.

Pero antes recordemos:

- Diferentes aplicaciones requieren diferentes conjuntos de servicios de seguridad. Por ejemplo, para los e-mail privado las tres primeras funciones son deseables, mientras que un sistema de correo electrónico de la empresa también puede requerir el no repudio.
- Otro ejemplo, es si queremos conseguir actualizaciones de software para un teléfono celular, los principales objetivos podría ser la integridad y autenticación de mensajes principalmente debido a que el fabricante quiere asegurar que sólo las actualizaciones originales se cargan en el dispositivo.
- Se debe observar que la autenticación de mensajes implica siempre la integridad de los datos, lo contrario no es cierto.
- Los cuatro servicios de seguridad pueden ser logrados de una manera más o menos directa con los algoritmos y protocolos que estudiaremos en este curso, Pro ejemplo:
  - Para la confidencialidad se utiliza principalmente sistemas de cifrado simétrico (AES, 3DES) y cifrado asimétrico con menos frecuencia (DLP, ECC, RSA).
  - La integridad y autenticación de mensajes son proporcionados por las firmas digitales y códigos de autenticación de mensajes (que estudiaremos más adelante en el curso).
  - No repudio se logra con las firmas digitales mencionados anteriormente.

Pero antes recordemos:

- Diferentes aplicaciones requieren diferentes conjuntos de servicios de seguridad. Por ejemplo, para los e-mail privado las tres primeras funciones son deseables, mientras que un sistema de correo electrónico de la empresa también puede requerir el no repudio.
- Otro ejemplo, es si queremos conseguir actualizaciones de software para un teléfono celular, los principales objetivos podría ser la integridad y autenticación de mensajes principalmente debido a que el fabricante quiere asegurar que sólo las actualizaciones originales se cargan en el dispositivo.
- Se debe observar que la autenticación de mensajes implica siempre la integridad de los datos, lo contrario no es cierto.
- Los cuatro servicios de seguridad pueden ser logrados de una manera más o menos directa con los algoritmos y protocolos que estudiaremos en este curso, Pro ejemplo:
  - Para la confidencialidad se utiliza principalmente sistemas de cifrado simétrico (AES, 3DES) y cifrado asimétrico con menos frecuencia (DLP, ECC, RSA).
  - La integridad y autenticación de mensajes son proporcionados por las firmas digitales y códigos de autenticación de mensajes (que estudiaremos más adelante en el curso).
  - No repudio se logra con las firmas digitales mencionados anteriormente.

Pero antes recordemos:

- Diferentes aplicaciones requieren diferentes conjuntos de servicios de seguridad. Por ejemplo, para los e-mail privado las tres primeras funciones son deseables, mientras que un sistema de correo electrónico de la empresa también puede requerir el no repudio.
- Otro ejemplo, es si queremos conseguir actualizaciones de software para un teléfono celular, los principales objetivos podría ser la integridad y autenticación de mensajes principalmente debido a que el fabricante quiere asegurar que sólo las actualizaciones originales se cargan en el dispositivo.
- Se debe observar que la autenticación de mensajes implica siempre la integridad de los datos, lo contrario no es cierto.
- Los cuatro servicios de seguridad pueden ser logrados de una manera más o menos directa con los algoritmos y protocolos que estudiaremos en este curso, Pro ejemplo:
  - Para la confidencialidad se utiliza principalmente sistemas de cifrado simétrico (AES, 3DES) y cifrado asimétrico con menos frecuencia (DLP, ECC, RSA).
  - La integridad y autenticación de mensajes son proporcionados por las firmas digitales y códigos de autenticación de mensajes (que estudiaremos más adelante en el curso).
  - No repudio se logra con las firmas digitales mencionados anteriormente.

Pero antes recordemos:

- Diferentes aplicaciones requieren diferentes conjuntos de servicios de seguridad. Por ejemplo, para los e-mail privado las tres primeras funciones son deseables, mientras que un sistema de correo electrónico de la empresa también puede requerir el no repudio.
- Otro ejemplo, es si queremos conseguir actualizaciones de software para un teléfono celular, los principales objetivos podría ser la integridad y autenticación de mensajes principalmente debido a que el fabricante quiere asegurar que sólo las actualizaciones originales se cargan en el dispositivo.
- Se debe observar que la autenticación de mensajes implica siempre la integridad de los datos, lo contrario no es cierto.
- Los cuatro servicios de seguridad pueden ser logrados de una manera más o menos directa con los algoritmos y protocolos que estudiaremos en este curso, Pro ejemplo:
  - Para la confidencialidad se utiliza principalmente sistemas de cifrado simétrico (AES, 3DES) y cifrado asimétrico con menos frecuencia (DLP, ECC, RSA).
  - La integridad y autenticación de mensajes son proporcionados por las firmas digitales y códigos de autenticación de mensajes (que estudiaremos más adelante en el curso).
  - No repudio se logra con las firmas digitales mencionados anteriormente.

Pero antes recordemos:

- Diferentes aplicaciones requieren diferentes conjuntos de servicios de seguridad. Por ejemplo, para los e-mail privado las tres primeras funciones son deseables, mientras que un sistema de correo electrónico de la empresa también puede requerir el no repudio.
- Otro ejemplo, es si queremos conseguir actualizaciones de software para un teléfono celular, los principales objetivos podría ser la integridad y autenticación de mensajes principalmente debido a que el fabricante quiere asegurar que sólo las actualizaciones originales se cargan en el dispositivo.
- Se debe observar que la autenticación de mensajes implica siempre la integridad de los datos, lo contrario no es cierto.
- Los cuatro servicios de seguridad pueden ser logrados de una manera más o menos directa con los algoritmos y protocolos que estudiaremos en este curso, Pro ejemplo:
  - Para la confidencialidad se utiliza principalmente sistemas de cifrado simétrico (AES, 3DES) y cifrado asimétrico con menos frecuencia (DLP, ECC, RSA).
  - La integridad y autenticación de mensajes son proporcionados por las firmas digitales y códigos de autenticación de mensajes (que estudiaremos más adelante en el curso).
  - No repudio se logra con las firmas digitales mencionados anteriormente.

Pero antes recordemos:

- Diferentes aplicaciones requieren diferentes conjuntos de servicios de seguridad. Por ejemplo, para los e-mail privado las tres primeras funciones son deseables, mientras que un sistema de correo electrónico de la empresa también puede requerir el no repudio.
- Otro ejemplo, es si queremos conseguir actualizaciones de software para un teléfono celular, los principales objetivos podría ser la integridad y autenticación de mensajes principalmente debido a que el fabricante quiere asegurar que sólo las actualizaciones originales se cargan en el dispositivo.
- Se debe observar que la autenticación de mensajes implica siempre la integridad de los datos, lo contrario no es cierto.
- Los cuatro servicios de seguridad pueden ser logrados de una manera más o menos directa con los algoritmos y protocolos que estudiaremos en este curso, Pro ejemplo:
  - Para la confidencialidad se utiliza principalmente sistemas de cifrado simétrico (AES, 3DES) y cifrado asimétrico con menos frecuencia (DLP, ECC, RSA).
  - La integridad y autenticación de mensajes son proporcionados por las firmas digitales y códigos de autenticación de mensajes (que estudiaremos más adelante en el curso).
  - No repudio se logra con las firmas digitales mencionados anteriormente.

Pero antes recordemos:

- Diferentes aplicaciones requieren diferentes conjuntos de servicios de seguridad. Por ejemplo, para los e-mail privado las tres primeras funciones son deseables, mientras que un sistema de correo electrónico de la empresa también puede requerir el no repudio.
- Otro ejemplo, es si queremos conseguir actualizaciones de software para un teléfono celular, los principales objetivos podría ser la integridad y autenticación de mensajes principalmente debido a que el fabricante quiere asegurar que sólo las actualizaciones originales se cargan en el dispositivo.
- Se debe observar que la autenticación de mensajes implica siempre la integridad de los datos, lo contrario no es cierto.
- Los cuatro servicios de seguridad pueden ser logrados de una manera más o menos directa con los algoritmos y protocolos que estudiaremos en este curso, Pro ejemplo:
  - Para la confidencialidad se utiliza principalmente sistemas de cifrado simétrico (AES, 3DES) y cifrado asimétrico con menos frecuencia (DLP, ECC, RSA).
  - La integridad y autenticación de mensajes son proporcionados por las firmas digitales y códigos de autenticación de mensajes (que estudiaremos más adelante en el curso).
  - No repudio se logra con las firmas digitales mencionados anteriormente.

Pero antes recordemos:

Además de los cuatro servicios de seguridad básicos hay varios otros:

Identificación / Autentificación de la entidad: Establecer y verificar la identidad de una entidad, por ejemplo, una persona, un computador o una smart cards.

Control de acceso: Restringir el acceso a los recursos.

Disponibilidad: Asegura que el sistema electrónico es disponible.

Auditoría: Presentar pruebas sobre las actividades relevantes para la seguridad, por ejemplo, al mantener registros acerca de ciertos acontecimientos.

Seguridad física: Proporcionar protección contra la manipulación física y/o respuestas a los intentos de manipulación física.

Anonimato: Proporcionar protección contra el descubrimiento y el uso indebido de la identidad.

The RSA Signature Scheme

- El esquema de firma RSA se basa en el cifrado RSA introducido anteriormente.
- Su seguridad se basa en la dificultad de factorizar un producto de dos números primos grandes (el problema de factorización de enteros).
- Desde su primera descripción en 1978, el esquema de firma RSA se ha convertido en el esquema de firmas digital más utilizado en la práctica.

The RSA Signature Scheme

- El esquema de firma RSA se basa en el cifrado RSA introducido anteriormente.
- Su seguridad se basa en la dificultad de factorizar un producto de dos números primos grandes (el problema de factorización de enteros).
- Desde su primera descripción en 1978, el esquema de firma RSA se ha convertido en el esquema de firmas digital más utilizado en la práctica.

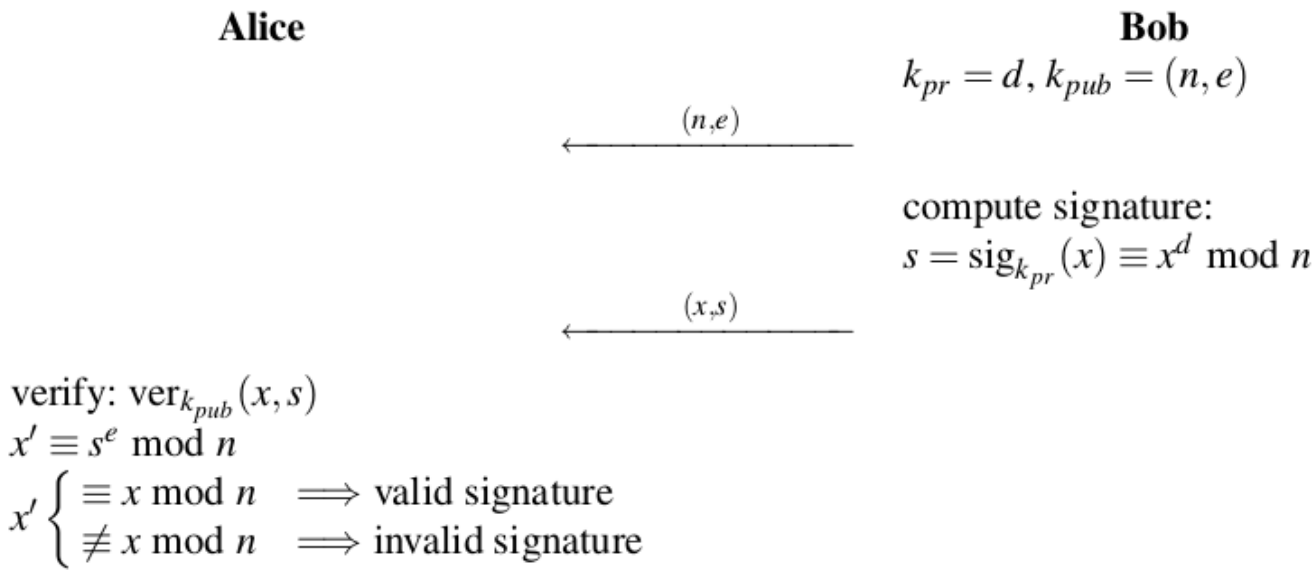
The RSA Signature Scheme

- El esquema de firma RSA se basa en el cifrado RSA introducido anteriormente.
- Su seguridad se basa en la dificultad de factorizar un producto de dos números primos grandes (el problema de factorización de enteros).
- Desde su primera descripción en 1978, el esquema de firma RSA se ha convertido en el esquema de firmas digital más utilizado en la práctica.

Schoolbook RSA Firma Digital

- Supongamos que Bob quiere enviar un mensaje x firmado a Alice.
- Se genera las mismas claves RSA que se utilizaron para el cifrado RSA. Es decir: RSA Keys
  - Bob's private key: $k_{pr} = (d)$.
  - Bob's public key: $k_{pub} = (n, e)$
- El protocolo de firma se muestra en la siguiente figura. El mensaje x que se está firmado pertenece al intervalo $\{1, 2, \ldots, n-1\}$ (Donde opera el RSA).

Basic RSA Digital Signature Protocol



Schoolbook RSA Firma Digital

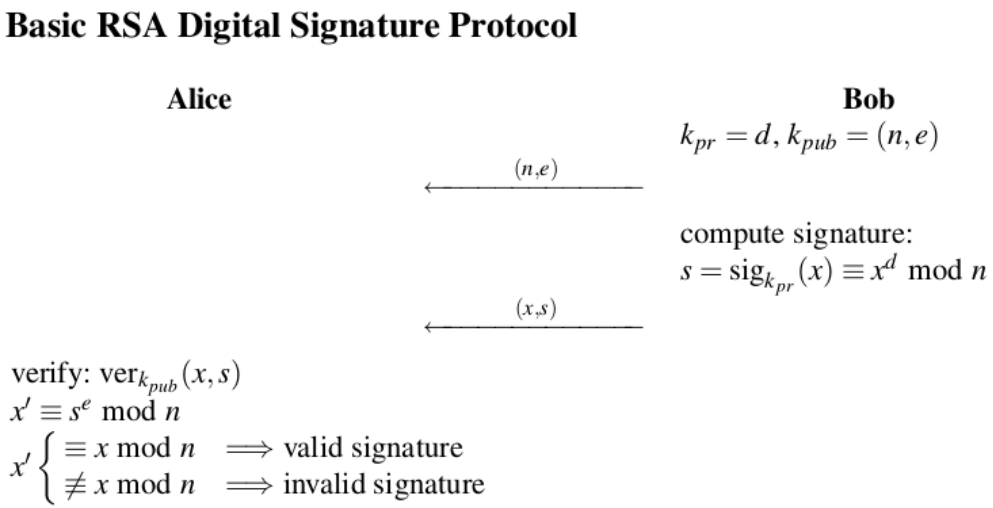


Figure: Basic RSA Digital Signature Protocol

- Como se puede ver a partir del protocolo, Bob calcula la firma s de un mensaje x utilizando el cifrado de RSA al mensaje x con su clave privada k_{pr}.
- Bob es el único que puede utilizar su clave privada k_{pr}, y por lo tanto k_{pr} lo autentifica como el autor de la firma del mensaje.
- Bob agrega la firma s al mensaje x y envía ambos a Alice.
- Alice por otro lado, recibe el mensaje firmado y utiliza el desifrador del RSA s utilizando la clave publica de Bob k_{pub}, obteniendo x.

Schoolbook RSA Firma Digital

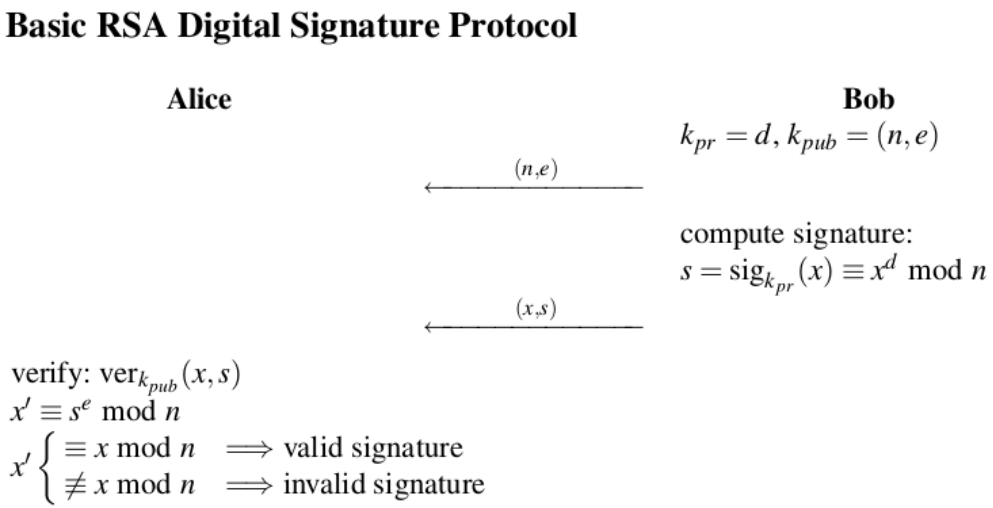


Figure: Basic RSA Digital Signature Protocol

- Como se puede ver a partir del protocolo, Bob calcula la firma s de un mensaje x utilizando el cifrado de RSA al mensaje x con su clave privada k_{pr}.
- Bob es el único que puede utilizar su clave privada k_{pr}, y por lo tanto k_{pr} lo autentifica como el autor de la firma del mensaje.
- Bob agrega la firma s al mensaje x y envía ambos a Alice.
- Alice por otro lado, recibe el mensaje firmado y utiliza el desifrador del RSA s utilizando la clave publica de Bob k_{pub}, obteniendo x.

Schoolbook RSA Firma Digital

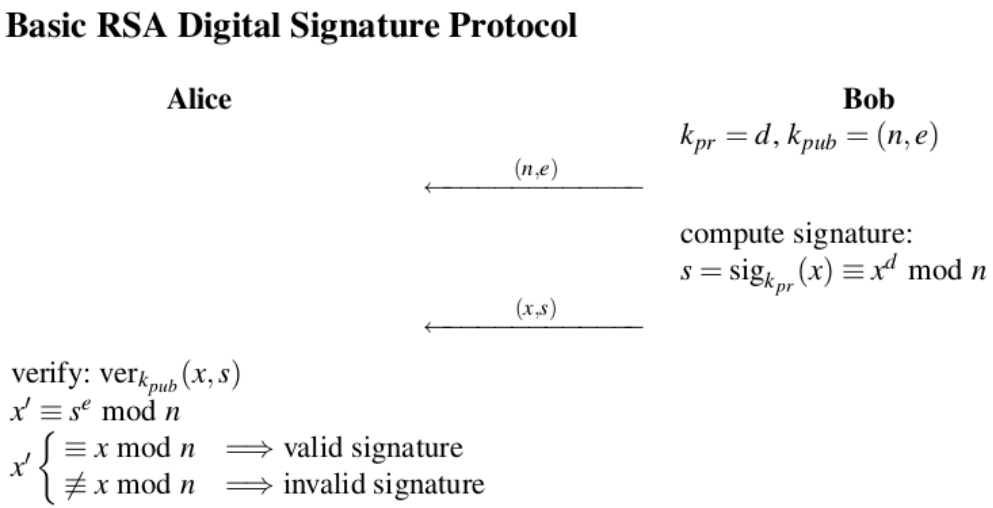


Figure: Basic RSA Digital Signature Protocol

- Como se puede ver a partir del protocolo, Bob calcula la firma s de un mensaje x utilizando el cifrado de RSA al mensaje x con su clave privada k_{pr}.
- Bob es el único que puede utilizar su clave privada k_{pr}, y por lo tanto k_{pr} lo autentifica como el autor de la firma del mensaje.
- Bob agrega la firma s al mensaje x y envía ambos a Alice.
- Alice por otro lado, recibe el mensaje firmado y utiliza el desifrador del RSA s utilizando la clave publica de Bob k_{pub}, obteniendo x.

Schoolbook RSA Firma Digital

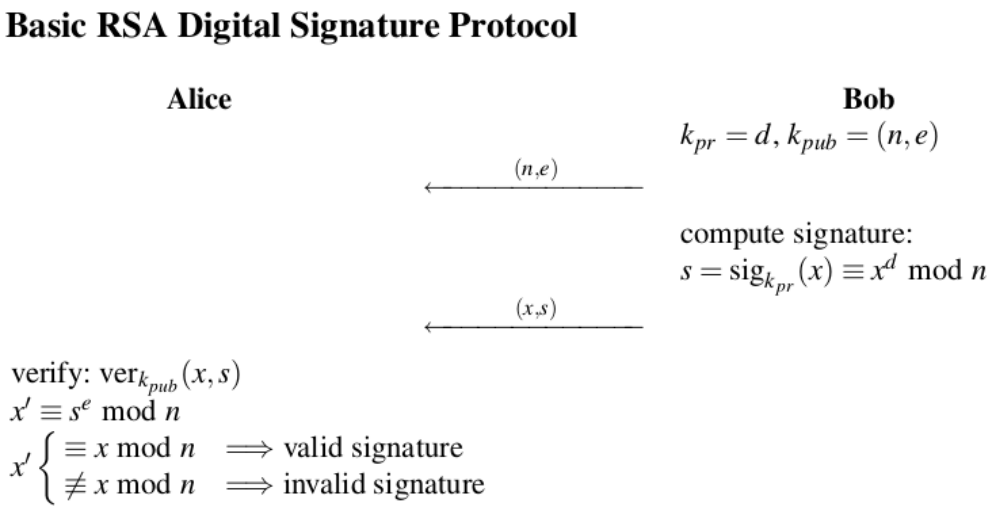


Figure: Basic RSA Digital Signature Protocol

- Como se puede ver a partir del protocolo, Bob calcula la firma s de un mensaje x utilizando el cifrado de RSA al mensaje x con su clave privada k_{pr}.
- Bob es el único que puede utilizar su clave privada k_{pr}, y por lo tanto k_{pr} lo autentifica como el autor de la firma del mensaje.
- Bob agrega la firma s al mensaje x y envía ambos a Alice.
- Alice por otro lado, recibe el mensaje firmado y utiliza el desifrador del RSA s utilizando la clave publica de Bob k_{pub}, obteniendo x.

Schoolbook RSA Firma Digital

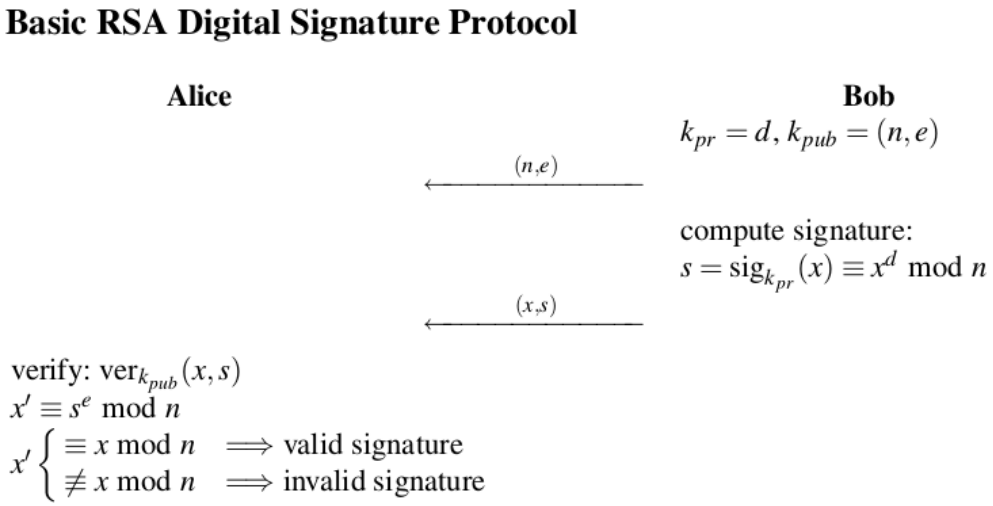


Figure: Basic RSA Digital Signature Protocol

- Si x y x' son iguales , Alice sabe dos cosas importantes:
  1. En primer lugar, el autor del mensaje estaba en posesión de la clave secreta de Bob k_{pr}, y si sólo Bob ha tenido acceso a la clave, luego fue Bob quien firmó el mensaje. Luego obtenemos *Autenticación de mensajes*.
  2. En segundo lugar, el mensaje no ha sido alterado durante el trayecto, por lo que se da la *Integridad del mensaje*.

Schoolbook RSA Firma Digital

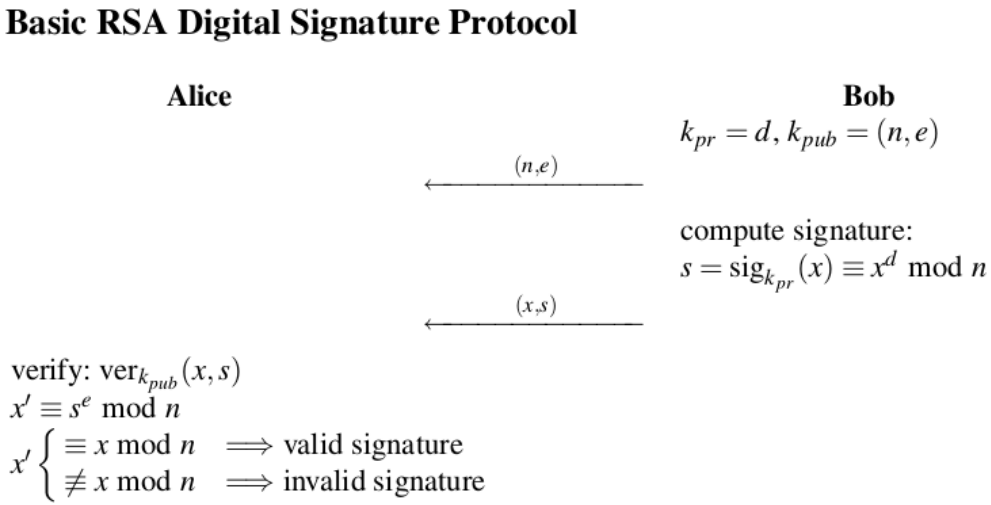


Figure: Basic RSA Digital Signature Protocol

- Si x y x' son iguales , Alice sabe dos cosas importantes:
  1. En primer lugar, el autor del mensaje estaba en posesión de la clave secreta de Bob k_{pr}, y si sólo Bob ha tenido acceso a la clave, luego fue Bob quien firmó el mensaje. Luego obtenemos *Autenticación de mensajes*.
  2. En segundo lugar, el mensaje no ha sido alterado durante el trayecto, por lo que se da la *Integridad del mensaje*.

Schoolbook RSA Firma Digital

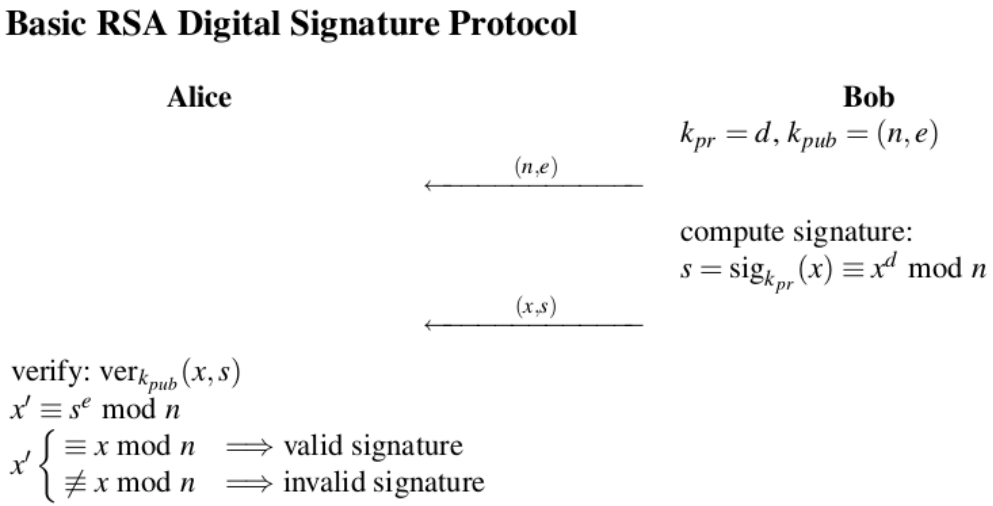


Figure: Basic RSA Digital Signature Protocol

- Si x y x' son iguales , Alice sabe dos cosas importantes:
  1. En primer lugar, el autor del mensaje estaba en posesión de la clave secreta de Bob k_{pr}, y si sólo Bob ha tenido acceso a la clave, luego fue Bob quien firmó el mensaje. Luego obtenemos *Autenticación de mensajes*.
  2. En segundo lugar, el mensaje no ha sido alterado durante el trayecto, por lo que se da la *Integridad del mensaje*.

Schoolbook RSA Firma Digital

Demostración

- Ahora demostraremos que el sistema es correcto, es decir, que el proceso de verificación produce una declaración "true" si el mensaje y la firma no han sido alterados durante la transmisión.
- Partimos de la operación de verificación:

$$s^e = (x^d)^e = x^{de} \equiv x \bmod n$$

- Dado la relación matemática entre la clave privada y la clave publica

$$de \equiv 1 \mod \phi(n)$$

elevar cualquier número entero $x \in \mathbb{Z}_n$ o la (de)-ésima potencia produce el mismo número entero de nuevo. La prueba de esto se dio anteriormente en el curso.

Schoolbook RSA Firma Digital

Demostración

- Ahora demostraremos que el sistema es correcto, es decir, que el proceso de verificación produce una declaración "true" si el mensaje y la firma no han sido alterados durante la transmisión.
- Partimos de la operación de verificación:

$$s^e = (x^d)^e = x^{de} \equiv x \bmod n$$

- Dado la relación matemática entre la clave privada y la clave publica

$$de \equiv 1 \mod \phi(n)$$

elevar cualquier número entero $x \in \mathbb{Z}_n$ o la (de)-ésima potencia produce el mismo número entero de nuevo. La prueba de esto se dio anteriormente en el curso.

Schoolbook RSA Firma Digital

Demostración

- Ahora demostraremos que el sistema es correcto, es decir, que el proceso de verificación produce una declaración "true" si el mensaje y la firma no han sido alterados durante la transmisión.
- Partimos de la operación de verificación:

$$s^e = (x^d)^e = x^{de} \equiv x \bmod n$$

- Dado la relación matemática entre la clave privada y la clave publica

$$de \equiv 1 \mod \phi(n)$$

elevar cualquier número entero $x \in \mathbb{Z}_n$ o la (de)-ésima potencia produce el mismo número entero de nuevo. La prueba de esto se dio anteriormente en el curso.

Schoolbook RSA Firma Digital

Observaciones

- *El rol de la clave pública y privada se intercambian al compararlo con el sistema de cifrado RSA.*
- *Recordemos que en el cifrado RSA se aplica la clave pública k_{pub} para el mensaje x, mientras que en el esquema de firma se aplica la clave privada k_{pr}.*
- *Por otro lado, del canal de comunicación, el cifrado RSA requiere el uso de la clave privada por el receptor, mientras que el esquema de firma digital se aplica la clave pública para la verificación.*

Schoolbook RSA Firma Digital

Observaciones

- *El rol de la clave pública y privada se intercambian al compararlo con el sistema de cifrado RSA.*
- *Recordemos que en el cifrado RSA se aplica la clave pública k_{pub} para el mensaje x, mientras que en el esquema de firma se aplica la clave privada k_{pr}.*
- *Por otro lado, del canal de comunicación, el cifrado RSA requiere el uso de la clave privada por el receptor, mientras que el esquema de firma digital se aplica la clave pública para la verificación.*

Schoolbook RSA Firma Digital

Observaciones

- *El rol de la clave pública y privada se intercambian al compararlo con el sistema de cifrado RSA.*
- *Recordemos que en el cifrado RSA se aplica la clave pública k_{pub} para el mensaje x, mientras que en el esquema de firma se aplica la clave privada k_{pr}.*
- *Por otro lado, del canal de comunicación, el cifrado RSA requiere el uso de la clave privada por el receptor, mientras que el esquema de firma digital se aplica la clave pública para la verificación.*

Schoolbook RSA Firma Digital

Veamos un ejemplo con números pequeños

- Supongamos que Bob quiere enviar un mensaje firmado ($x = 4$) a Alice.
- Los primeros pasos son exactamente los mismos como se hace para un cifrado RSA:
  - Bob calcula sus parámetros de RSA y envía la clave pública a Alice. En contraste con el esquema de cifrado, ahora la clave privada se utiliza para firmar, mientras que se necesita la clave pública para verificar la firma.

Alice		**Bob**
		1. choose $p = 3$ and $q = 11$ 2. $n = p \cdot q = 33$ 3. $\Phi(n) = (3-1)(11-1) = 20$ 4. choose $e = 3$ 5. $d \equiv e^{-1} \equiv 7 \bmod 20$
	$\xleftarrow{(n,e)=(33,3)}$	
		compute signature for message $x = 4$: $s = x^d \equiv 4^7 \equiv 16 \bmod 33$
	$\xleftarrow{(x,s)=(4,16)}$	
verify: $x' = s^e \equiv 16^3 \equiv 4 \bmod 33$ $x' \equiv x \bmod 33 \Longrightarrow$ valid signature		

Schoolbook RSA Firma Digital

Veamos un ejemplo con números pequeños

- Alice, puede concluir que dado la firma válida que Bob a generado el mensaje
- y que no se ha alterado el mensajo en tránsito, es decir, la autenticación de mensajes y la integridad del mensaje.

Schoolbook RSA Firma Digital

Observación:

- Cabe señalar que se introdujo un esquema de firma digital solamente.
- En particular, el mensaje no está cifrado y, por lo tanto, no hay confidencialidad.
- Si se requiere este servicio de seguridad, el mensaje junto con la firma debe ser encriptada,
- por ejemplo, utilizando un algoritmo simétrico como AES.

Schoolbook RSA Firma Digital

Observación:

- Cabe señalar que se introdujo un esquema de firma digital solamente.
- En particular, el mensaje no está cifrado y, por lo tanto, no hay confidencialidad.
- Si se requiere este servicio de seguridad, el mensaje junto con la firma debe ser encriptada,
- por ejemplo, utilizando un algoritmo simétrico como AES.

Schoolbook RSA Firma Digital

Observación:

- Cabe señalar que se introdujo un esquema de firma digital solamente.
- En particular, el mensaje no está cifrado y, por lo tanto, no hay confidencialidad.
- Si se requiere este servicio de seguridad, el mensaje junto con la firma debe ser encriptada,
- por ejemplo, utilizando un algoritmo simétrico como AES.

Schoolbook RSA Firma Digital

Observación:

- Cabe señalar que se introdujo un esquema de firma digital solamente.
- En particular, el mensaje no está cifrado y, por lo tanto, no hay confidencialidad.
- Si se requiere este servicio de seguridad, el mensaje junto con la firma debe ser encriptada,
- por ejemplo, utilizando un algoritmo simétrico como AES.

Schoolbook RSA Firma Digital

Ataque

Existential Forgery Attack Against RSA Digital Signature

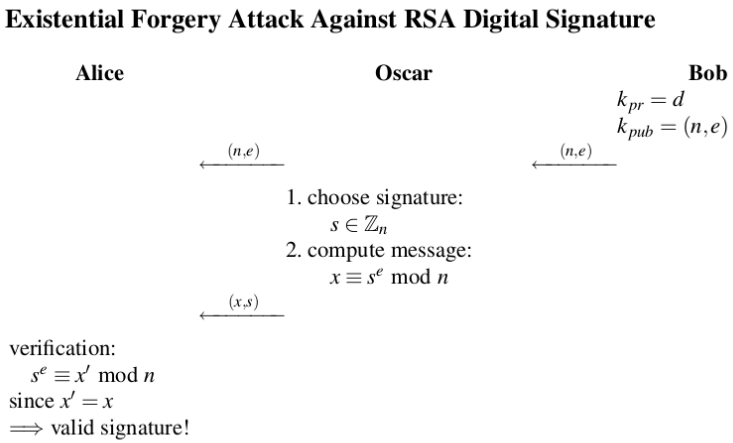
Alice	Oscar	Bob
		$k_{pr} = d$ $k_{pub} = (n,e)$
	$\xleftarrow{(n,e)}$	$\xleftarrow{(n,e)}$
	1. choose signature: $s \in \mathbb{Z}_n$ 2. compute message: $x \equiv s^e \bmod n$	
	$\xleftarrow{(x,s)}$	

verification:
$s^e \equiv x' \bmod n$
since $x' = x$
$\Longrightarrow$ valid signature!

- El atacante suplanta a Bob, es decir, Oscar dice a Alice que él es Bob.

Schoolbook RSA Firma Digital

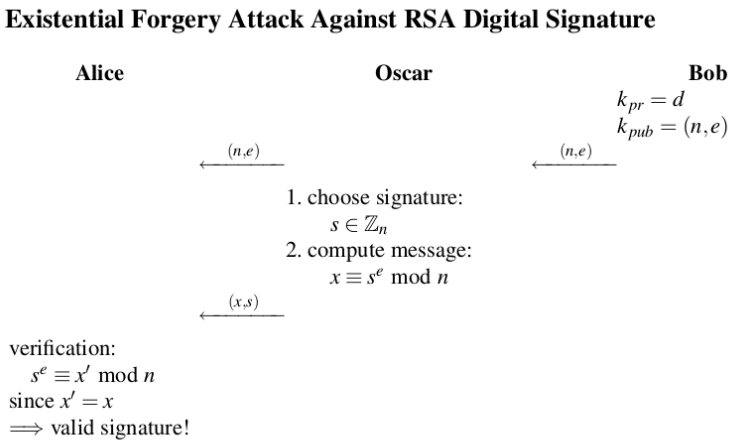
Ataque



- Alice realiza exactamente los mismos cálculos que Oscar, y se verificará la firma como correcta.
- Sin embargo, por estrechamente mirando a los pasos 1 y 2 que Oscar realiza, se ve que el ataque es un poco extraño.
- El atacante elige la firma primero y luego calcula el mensaje. Como consecuencia, no puede controlar la semántica del mensaje de x.

Schoolbook RSA Firma Digital

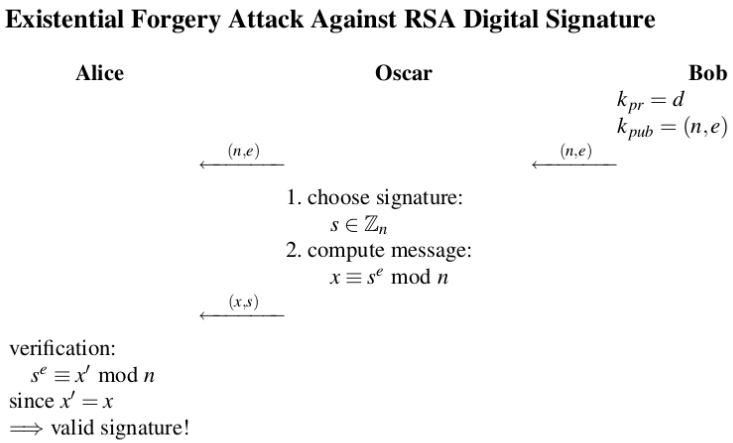
Ataque



- Alice realiza exactamente los mismos cálculos que Oscar, y se verificará la firma como correcta.
- Sin embargo, por estrechamente mirando a los pasos 1 y 2 que Oscar realiza, se ve que el ataque es un poco extraño.
- El atacante elige la firma primero y luego calcula el mensaje. Como consecuencia, no puede controlar la semántica del mensaje de x.

Schoolbook RSA Firma Digital

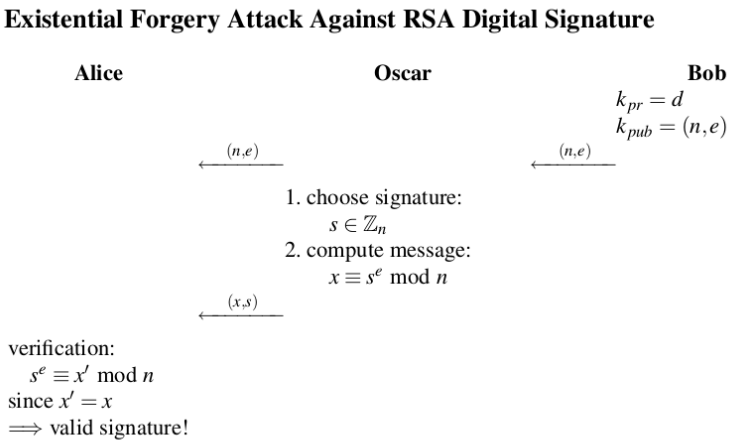
Ataque



- Alice realiza exactamente los mismos cálculos que Oscar, y se verificará la firma como correcta.
- Sin embargo, por estrechamente mirando a los pasos 1 y 2 que Oscar realiza, se ve que el ataque es un poco extraño.
- El atacante elige la firma primero y luego calcula el mensaje. Como consecuencia, no puede controlar la semántica del mensaje de x.

Schoolbook RSA Firma Digital

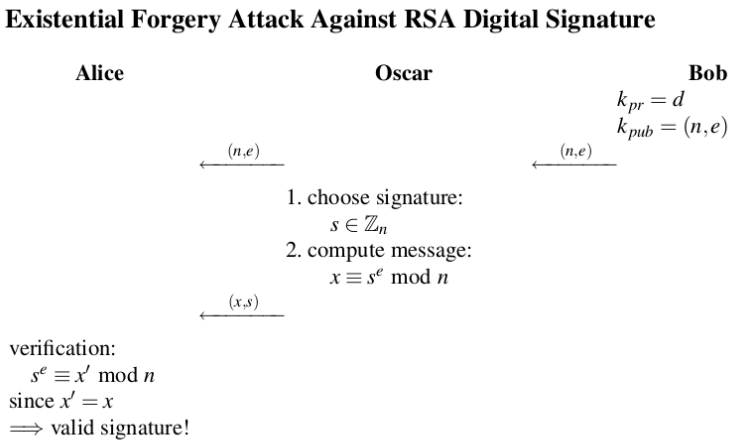
Ataque



- Por ejemplo, Oscar no puede generar un mensaje del tipo "Transfer $1000 en la cuenta del Oscar".
- Sin embargo, el hecho de que un proceso de verificación automatizada no reconoce la falsificación no es una característica muy deseable.
- Por esta razón, *schoolbook RSA firma Digital* se utiliza raramente en la práctica, y se aplican esquemas de relleno con el fin de evitar que este y otros tipos de ataques.

Schoolbook RSA Firma Digital

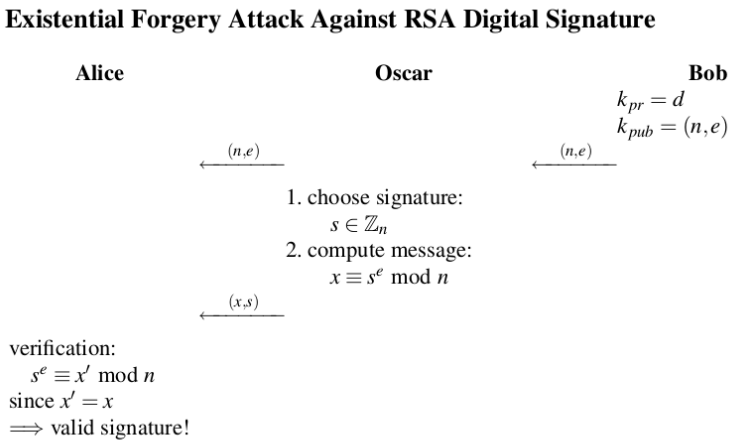
Ataque



- Por ejemplo, Oscar no puede generar un mensaje del tipo "Transfer $1000 en la cuenta del Oscar".
- Sin embargo, el hecho de que un proceso de verificación automatizada no reconoce la falsificación no es una característica muy deseable.
- Por esta razón, *schoolbook RSA firma Digital* se utiliza raramente en la práctica, y se aplican esquemas de relleno con el fin de evitar que este y otros tipos de ataques.

Schoolbook RSA Firma Digital

Ataque



- Por ejemplo, Oscar no puede generar un mensaje del tipo "Transfer $1000 en la cuenta del Oscar".
- Sin embargo, el hecho de que un proceso de verificación automatizada no reconoce la falsificación no es una característica muy deseable.
- Por esta razón, *schoolbook RSA firma Digital* se utiliza raramente en la práctica, y se aplican esquemas de relleno con el fin de evitar que este y otros tipos de ataques.

RSA Padding: The Probabilistic Signature Standard (PSS)

- El ataque anteriormente se pueden prevenir al permitir sólo ciertos formatos de mensaje.
- En términos generales, el formato impone una norma que permite al verificador, Alice, distinguir entre los mensajes válidos e inválidos, lo que se llama *padding*.
- Por ejemplo, una simple regla de formato, podría especificar que todos los mensajes x tienen 100 bits de arrastre con el valor cero (o cualquier otro patrón específico bit).
- Si Oscar escoge valores de firma s y calcula el "mensaje" $x \equiv s^e \bmod n$, es poco probable que x tiene este formato específico.
- Si se requiere un cierto valor para los 100 bits de cola, la probabilidad de que x tenga este formato es de 2^{-100}, que es una probabilidad muy pequeña.

RSA Padding: The Probabilistic Signature Standard (PSS)

- El ataque anteriormente se pueden prevenir al permitir sólo ciertos formatos de mensaje.
- En términos generales, el formato impone una norma que permite al verificador, Alice, distinguir entre los mensajes válidos e inválidos, lo que se llama *padding*.
- Por ejemplo, una simple regla de formato, podría especificar que todos los mensajes x tienen 100 bits de arrastre con el valor cero (o cualquier otro patrón específico bit).
- Si Oscar escoge valores de firma s y calcula el "mensaje" $x \equiv s^e \bmod n$, es poco probable que x tiene este formato específico.
- Si se requiere un cierto valor para los 100 bits de cola, la probabilidad de que x tenga este formato es de 2^{-100}, que es una probabilidad muy pequeña.

RSA Padding: The Probabilistic Signature Standard (PSS)

- El ataque anteriormente se pueden prevenir al permitir sólo ciertos formatos de mensaje.
- En términos generales, el formato impone una norma que permite al verificador, Alice, distinguir entre los mensajes válidos e inválidos, lo que se llama *padding*.
- Por ejemplo, una simple regla de formato, podría especificar que todos los mensajes x tienen 100 bits de arrastre con el valor cero (o cualquier otro patrón específico bit).
- Si Oscar escoge valores de firma s y calcula el "mensaje" $x \equiv s^e \bmod n$, es poco probable que x tiene este formato específico.
- Si se requiere un cierto valor para los 100 bits de cola, la probabilidad de que x tenga este formato es de 2^{-100}, que es una probabilidad muy pequeña.

RSA Padding: The Probabilistic Signature Standard (PSS)

- El ataque anteriormente se pueden prevenir al permitir sólo ciertos formatos de mensaje.
- En términos generales, el formato impone una norma que permite al verificador, Alice, distinguir entre los mensajes válidos e inválidos, lo que se llama *padding*.
- Por ejemplo, una simple regla de formato, podría especificar que todos los mensajes x tienen 100 bits de arrastre con el valor cero (o cualquier otro patrón específico bit).
- Si Oscar escoge valores de firma s y calcula el "mensaje" $x \equiv s^e \bmod n$, es poco probable que x tiene este formato específico.
- Si se requiere un cierto valor para los 100 bits de cola, la probabilidad de que x tenga este formato es de 2^{-100}, que es una probabilidad muy pequeña.

RSA Padding: The Probabilistic Signature Standard (PSS)

- El ataque anteriormente se pueden prevenir al permitir sólo ciertos formatos de mensaje.
- En términos generales, el formato impone una norma que permite al verificador, Alice, distinguir entre los mensajes válidos e inválidos, lo que se llama *padding*.
- Por ejemplo, una simple regla de formato, podría especificar que todos los mensajes x tienen 100 bits de arrastre con el valor cero (o cualquier otro patrón específico bit).
- Si Oscar escoge valores de firma s y calcula el "mensaje" $x \equiv s^e \bmod n$, es poco probable que x tiene este formato específico.
- Si se requiere un cierto valor para los 100 bits de cola, la probabilidad de que x tenga este formato es de 2^{-100}, que es una probabilidad muy pequeña.

RSA Padding: The Probabilistic Signature Standard (PSS)

- Un esquema de relleno que se utiliza ampliamente en la práctica. Es el esquema de firma probabilístico (RSA-PSS) es un esquema de firma con base en el sistema de cifrado RSA.
- Se combina la firma y la verificación con una codificación del mensaje.

RSA Padding: The Probabilistic Signature Standard (PSS)

- Un esquema de relleno que se utiliza ampliamente en la práctica. Es el esquema de firma probabilístico (RSA-PSS) es un esquema de firma con base en el sistema de cifrado RSA.
- Se combina la firma y la verificación con una codificación del mensaje.

Observación

- *Casi siempre en la práctica, el mensaje en sí mismo no está firmado directamente sino más bien la versión hash del mensaje.*
- *Las funciones hash para calcular una firma digital de mensajes, tiene una longitud fija, de 160 o 256-bits, pero acepta mensajes como entradas de longitudes arbitrarias.*
- *Conocer las funciones de hash y el papel en la firma digital lo estudiaremos más adelante en este curso.*

Observación

- *Casi siempre en la práctica, el mensaje en sí mismo no está firmado directamente sino más bien la versión hash del mensaje.*
- *Las funciones hash para calcular una firma digital de mensajes, tiene una longitud fija, de* 160 *o* 256*-bits, pero acepta mensajes como entradas de longitudes arbitrarias.*
- *Conocer las funciones de hash y el papel en la firma digital lo estudiaremos más adelante en este curso.*

Observación

- *Casi siempre en la práctica, el mensaje en sí mismo no está firmado directamente sino más bien la versión hash del mensaje.*
- *Las funciones hash para calcular una firma digital de mensajes, tiene una longitud fija, de* 160 *o* 256*-bits, pero acepta mensajes como entradas de longitudes arbitrarias.*
- *Conocer las funciones de hash y el papel en la firma digital lo estudiaremos más adelante en este curso.*

- Con el fin de mantener la coherencia con la terminología empleada en los standars, se denota la mensaje con M en lugar de x.
- En la siguiente figura, muestra el procedimiento de codificación, lo que se conoce como *Método de codificación de firma con apéndice (EMSA) Esquema Firma probabilístico (PSS)*.

Encoding for the EMSA Probabilistic Signature Scheme

Let $|n|$ be the size of the RSA modulus in bits. The encoded message EM has a length $\lceil (|n|-1)/8 \rceil$ bytes such that the bit length of EM is at most $|n|-1$ bit.

1. Generate a random value $salt$.
2. Form a string M' by concatenating a fixed padding $padding_1$, the hash value $mHash = h(M)$ and $salt$.
3. Compute a hash value H of the string M'.
4. Concatenate a fixed padding $padding_2$ and the value $salt$ to form a data block DB.
5. Apply a mask generation function MGF to the string M' to compute the mask value $dbMask$. In practice, a hash function such as SHA-1 is often used as MGF.
6. XOR the mask value $dbMask$ and the data block DB to compute $maskedDB$.
7. The encoded message EM is obtained by concatenating $maskedDB$, the hash value H and the fixed padding bc.

Encoding for the EMSA Probabilistic Signature Scheme

Let $|n|$ be the size of the RSA modulus in bits. The encoded message EM has a length $\lceil (|n|-1)/8 \rceil$ bytes such that the bit length of EM is at most $|n|-1$ bit.

1. Generate a random value $salt$.
2. Form a string M' by concatenating a fixed padding $padding_1$, the hash value $mHash = h(M)$ and $salt$.
3. Compute a hash value H of the string M'.
4. Concatenate a fixed padding $padding_2$ and the value $salt$ to form a data block DB.
5. Apply a mask generation function MGF to the string M' to compute the mask value $dbMask$. In practice, a hash function such as SHA-1 is often used as MGF.
6. XOR the mask value $dbMask$ and the data block DB to compute $maskedDB$.
7. The encoded message EM is obtained by concatenating $maskedDB$, the hash value H and the fixed padding bc.

Después de la codificación, la operación real de firma se aplica a los EM de mensajes codificados, por ejemplo,

$$s = sig_{k_{pr}}(x) \equiv EM^d \bmod n$$

- La operación de verificación se procede de una manera similar: la recuperación del valor de *salt* y comprobar si la codificación EMSA-PSS del mensaje es correcto.
- Tenga en cuenta que el receptor conoce los valores de $padding_1$ y $padding_2$ del standard.
- El valor de H en EM es en esencia la versión con hash del mensaje (que estudiaremos más adelante en el curso).
- Mediante la adición del valor aleatorio *"sal"* antes del segundo hash, el valor codificado se convierte en probabilístico.
- En consecuencia, si codificamos y firmar el mismo mensaje dos veces, obtenemos diferentes firmas, lo cual es una característica deseable.

- La operación de verificación se procede de una manera similar: la recuperación del valor de *salt* y comprobar si la codificación EMSA-PSS del mensaje es correcto.
- Tenga en cuenta que el receptor conoce los valores de $padding_1$ y $padding_2$ del standard.
- El valor de H en EM es en esencia la versión con hash del mensaje (que estudiaremos más adelante en el curso).
- Mediante la adición del valor aleatorio *"sal"* antes del segundo hash, el valor codificado se convierte en probabilístico.
- En consecuencia, si codificamos y firmar el mismo mensaje dos veces, obtenemos diferentes firmas, lo cual es una característica deseable.

- La operación de verificación se procede de una manera similar: la recuperación del valor de *salt* y comprobar si la codificación EMSA-PSS del mensaje es correcto.
- Tenga en cuenta que el receptor conoce los valores de $padding_1$ y $padding_2$ del standard.
- El valor de H en EM es en esencia la versión con hash del mensaje (que estudiaremos más adelante en el curso).
- Mediante la adición del valor aleatorio *"sal"* antes del segundo hash, el valor codificado se convierte en probabilístico.
- En consecuencia, si codificamos y firmar el mismo mensaje dos veces, obtenemos diferentes firmas, lo cual es una característica deseable.

- La operación de verificación se procede de una manera similar: la recuperación del valor de *salt* y comprobar si la codificación EMSA-PSS del mensaje es correcto.
- Tenga en cuenta que el receptor conoce los valores de $padding_1$ y $padding_2$ del standard.
- El valor de H en EM es en esencia la versión con hash del mensaje (que estudiaremos más adelante en el curso).
- Mediante la adición del valor aleatorio *"sal"* antes del segundo hash, el valor codificado se convierte en probabilístico.
- En consecuencia, si codificamos y firmar el mismo mensaje dos veces, obtenemos diferentes firmas, lo cual es una característica deseable.

- La operación de verificación se procede de una manera similar: la recuperación del valor de *salt* y comprobar si la codificación EMSA-PSS del mensaje es correcto.
- Tenga en cuenta que el receptor conoce los valores de $padding_1$ y $padding_2$ del standard.
- El valor de H en EM es en esencia la versión con hash del mensaje (que estudiaremos más adelante en el curso).
- Mediante la adición del valor aleatorio *"sal"* antes del segundo hash, el valor codificado se convierte en probabilístico.
- En consecuencia, si codificamos y firmar el mismo mensaje dos veces, obtenemos diferentes firmas, lo cual es una característica deseable.

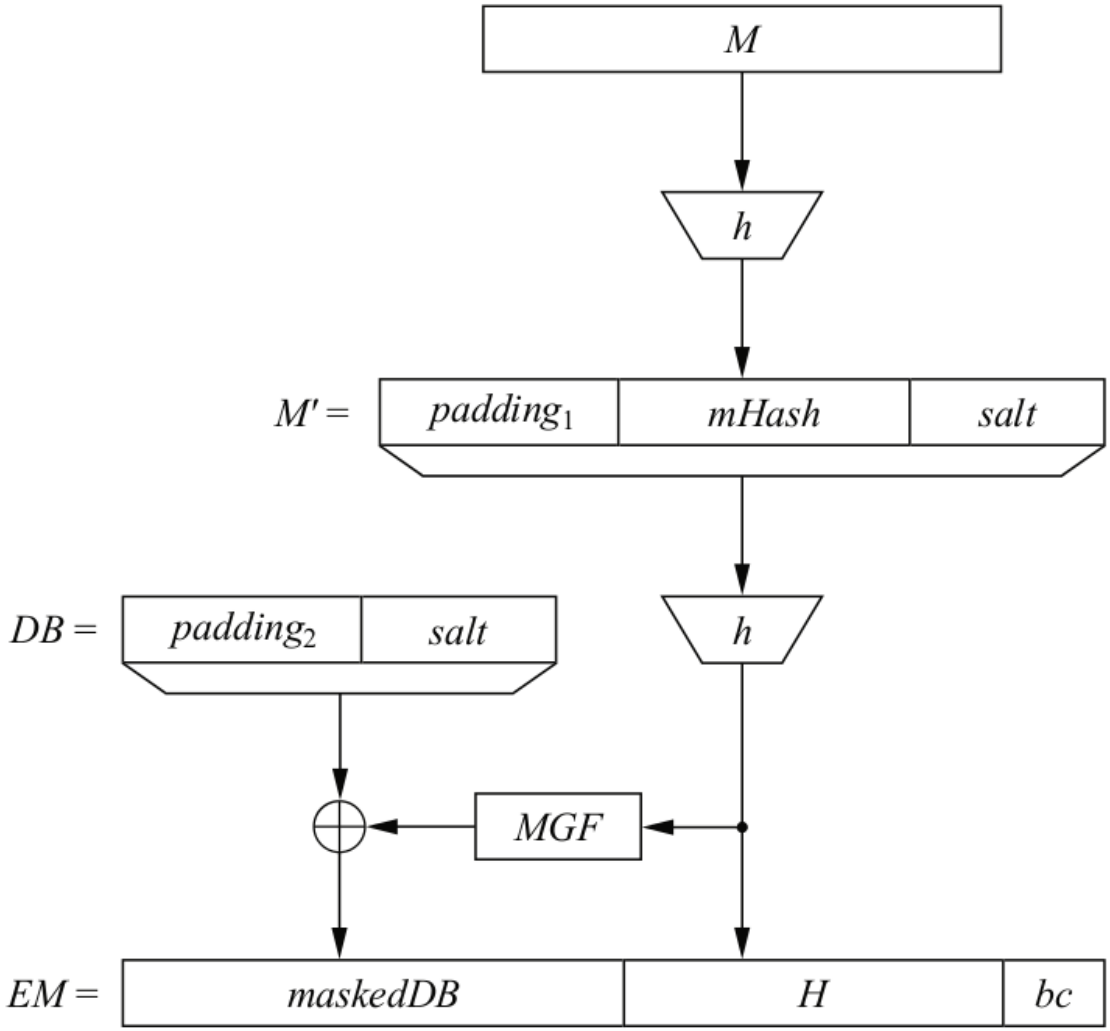


Figure: Principle of EMSA-PSS encoding

The Digital Signature Algorithm (DSA)

- En la práctica se utiliza el algoritmo de firma digital *Digital Signature Algorithm* (DSA).
- Es un estándar del gobierno federal de los EE.UU. para la firma digital (DSS) y fue propuesto por el Instituto Nacional de Estándares y Tecnología (NIST).
- Sus principales ventajas sobre el esquema de firma ElGamal son que la firma es sólo de 320-bits de largo y que algunos de los ataques que pueden poner en peligro el sistema Elgamal no son aplicables.

The Digital Signature Algorithm (DSA)

- En la práctica se utiliza el algoritmo de firma digital *Digital Signature Algorithm* (DSA).
- Es un estándar del gobierno federal de los EE.UU. para la firma digital (DSS) y fue propuesto por el Instituto Nacional de Estándares y Tecnología (NIST).
- Sus principales ventajas sobre el esquema de firma ElGamal son que la firma es sólo de 320-bits de largo y que algunos de los ataques que pueden poner en peligro el sistema Elgamal no son aplicables.

The Digital Signature Algorithm (DSA)

- En la práctica se utiliza el algoritmo de firma digital *Digital Signature Algorithm* (DSA).
- Es un estándar del gobierno federal de los EE.UU. para la firma digital (DSS) y fue propuesto por el Instituto Nacional de Estándares y Tecnología (NIST).
- Sus principales ventajas sobre el esquema de firma ElGamal son que la firma es sólo de 320-bits de largo y que algunos de los ataques que pueden poner en peligro el sistema Elgamal no son aplicables.

The DSA Algorithm

Presentamos aquí el estándar DSA con una longitud de 1024 bits. Longitudes de bits más largos también son posibles en el estándar.

Key Generation for DSA

1. Generate a prime p with $2^{1023} < p < 2^{1024}$.
2. Find a prime divisor q of $p-1$ with $2^{159} < q < 2^{160}$.
3. Find an element α with $\text{ord}(\alpha) = q$, i.e., α generates the subgroup with q elements.
4. Choose a random integer d with $0 < d < q$.
5. Compute $\beta \equiv \alpha^d \bmod p$.

The keys are now:

$k_{pub} = (p, q, \alpha, \beta)$

$k_{pr} = (d)$

Figure: Key Generation for DSA

The DSA Algorithm

- La idea central de DSA es que hay dos grupos cíclicos que intervienen.
- Uno de ellos es el grupo cíclico $\mathbb{Z}_p^*$, cuyo orden es de longitud de bits de 1024-bits.
- El segundo es un subgrupo de 160-bits de $\mathbb{Z}_p^*$. Esta configuración resulta en firmas más cortas.
- Además del primo p de 1024-bits y un primo q de 160-bits, hay otros dos posibles combinaciones de longitudes de bits posibles para los números primos p y q.
- De acuerdo a la última versión del estandar, se permite que las combinaciones que se muestran en la siguiente Tabla.
- Si se requiere alguna de las otras longitudes de bits, sólo los pasos 1 y 2 de la generación de claves fase tiene que ajustarse.

Table:

p	q	Signature
1024	160	320
2048	224	448
3072	256	512

The DSA Algorithm

- La idea central de DSA es que hay dos grupos cíclicos que intervienen.
- Uno de ellos es el grupo cíclico $\mathbb{Z}_p^*$, cuyo orden es de longitud de bits de 1024-bits.
- El segundo es un subgrupo de 160-bits de $\mathbb{Z}_p^*$. Esta configuración resulta en firmas más cortas.
- Además del primo p de 1024-bits y un primo q de 160-bits, hay otros dos posibles combinaciones de longitudes de bits posibles para los números primos p y q.
- De acuerdo a la última versión del estandar, se permite que las combinaciones que se muestran en la siguiente Tabla.
- Si se requiere alguna de las otras longitudes de bits, sólo los pasos 1 y 2 de la generación de claves fase tiene que ajustarse.

Table:

p	q	Signature
1024	160	320
2048	224	448
3072	256	512

The DSA Algorithm

- La idea central de DSA es que hay dos grupos cíclicos que intervienen.
- Uno de ellos es el grupo cíclico $\mathbb{Z}_p^*$, cuyo orden es de longitud de bits de 1024-bits.
- El segundo es un subgrupo de 160-bits de $\mathbb{Z}_p^*$. Esta configuración resulta en firmas más cortas.
- Además del primo p de 1024-bits y un primo q de 160-bits, hay otros dos posibles combinaciones de longitudes de bits posibles para los números primos p y q.
- De acuerdo a la última versión del estandar, se permite que las combinaciones que se muestran en la siguiente Tabla.
- Si se requiere alguna de las otras longitudes de bits, sólo los pasos 1 y 2 de la generación de claves fase tiene que ajustarse.

Table:

p	q	Signature
1024	160	320
2048	224	448
3072	256	512

The DSA Algorithm

- La idea central de DSA es que hay dos grupos cíclicos que intervienen.
- Uno de ellos es el grupo cíclico $\mathbb{Z}_p^*$, cuyo orden es de longitud de bits de 1024-bits.
- El segundo es un subgrupo de 160-bits de $\mathbb{Z}_p^*$. Esta configuración resulta en firmas más cortas.
- Además del primo p de 1024-bits y un primo q de 160-bits, hay otros dos posibles combinaciones de longitudes de bits posibles para los números primos p y q.
- De acuerdo a la última versión del estandar, se permite que las combinaciones que se muestran en la siguiente Tabla.
- Si se requiere alguna de las otras longitudes de bits, sólo los pasos 1 y 2 de la generación de claves fase tiene que ajustarse.

Table:

p	q	Signature
1024	160	320
2048	224	448
3072	256	512

The DSA Algorithm

- La idea central de DSA es que hay dos grupos cíclicos que intervienen.
- Uno de ellos es el grupo cíclico $\mathbb{Z}_p^*$, cuyo orden es de longitud de bits de 1024-bits.
- El segundo es un subgrupo de 160-bits de $\mathbb{Z}_p^*$. Esta configuración resulta en firmas más cortas.
- Además del primo p de 1024-bits y un primo q de 160-bits, hay otros dos posibles combinaciones de longitudes de bits posibles para los números primos p y q.
- De acuerdo a la última versión del estandar, se permite que las combinaciones que se muestran en la siguiente Tabla.
- Si se requiere alguna de las otras longitudes de bits, sólo los pasos 1 y 2 de la generación de claves fase tiene que ajustarse.

Table:

p	q	Signature
1024	160	320
2048	224	448
3072	256	512

The DSA Algorithm

- La idea central de DSA es que hay dos grupos cíclicos que intervienen.
- Uno de ellos es el grupo cíclico $\mathbb{Z}_p^*$, cuyo orden es de longitud de bits de 1024-bits.
- El segundo es un subgrupo de 160-bits de $\mathbb{Z}_p^*$. Esta configuración resulta en firmas más cortas.
- Además del primo p de 1024-bits y un primo q de 160-bits, hay otros dos posibles combinaciones de longitudes de bits posibles para los números primos p y q.
- De acuerdo a la última versión del estandar, se permite que las combinaciones que se muestran en la siguiente Tabla.
- Si se requiere alguna de las otras longitudes de bits, sólo los pasos 1 y 2 de la generación de claves fase tiene que ajustarse.

Table:

p	q	Signature
1024	160	320
2048	224	448
3072	256	512

Firma y Verificación

- La firma DSA consiste en un par de enteros (r, s).
- Luego cada uno de los dos parámetros es de sólo 160-bits de largo, la longitud total de la firma es de 320-bits.
- Usando la clave pública y la clave privada, la firma de un mensaje x se calcula de la siguiente manera:

DSA Signature Generation

1. Choose an integer as random ephemeral key k_E with $0 < k_E < q$.
2. Compute $r \equiv (\alpha^{k_E} \bmod p) \bmod q$.
3. Compute $s \equiv (SHA(x) + d \cdot r)\, {k_E}^{-1} \bmod q$.

Figure: DSA Signature Generation

Firma y Verificación

DSA Signature Generation

1. Choose an integer as random ephemeral key k_E with $0 < k_E < q$.
2. Compute $r \equiv (\alpha^{k_E} \bmod p) \bmod q$.
3. Compute $s \equiv (SHA(x) + d \cdot r)\, {k_E}^{-1} \bmod q$.

Figure: DSA Signature Generation

- De acuerdo con el estándar, el mensaje x tiene que ser hash usando la función hash SHA-1 con el fin de calcular s.
- Las funciones hash, como SHA-1, se describen más adelante en este curso.
- Por ahora es suficiente saber que SHA-1 comprime x y calcula una huella digital de 160-bits.
- Esta huella digital puede ser pensado como un representante de x.

Firma y Verificación

El proceso de verificación de la firma es la siguiente:

DSA Signature Verification

1. Compute auxiliary value $w \equiv s^{-1} \bmod q$.
2. Compute auxiliary value $u_1 \equiv w \cdot SHA(x) \bmod q$.
3. Compute auxiliary value $u_2 \equiv w \cdot r \bmod q$.
4. Compute $v \equiv (\alpha^{u_1} \cdot \beta^{u_2} \bmod p) \bmod q$.
5. The verification $ver_{k_{pub}}(x,(r,s))$ follows from:

$$v \begin{cases} \equiv r \bmod q \Longrightarrow \text{valid signature} \\ \not\equiv r \bmod q \Longrightarrow \text{invalid signature} \end{cases}$$

Figure: DSA Signature Verification

- El verificador acepta una firma (r,s) sólo si $v \equiv r \bmod q$ se cumple.
- De lo contrario,la verificación falla.
- En este caso, el mensaje o la firma pueden haber sido modificados.
- O el verificador no está en posesión de la clave pública correcta. En cualquier caso, la firma debe ser considerado no válido.

Demostración

Mostraremos que la firma (r, s) satisface la condición de verificación

$$\text{Pd:} \quad v \equiv r \bmod q$$

Comencemos con el parametro de firma s:

$$s \equiv (SHA(x) + dr)k_E^{-1} \bmod q$$

Luego, existe $j \in \mathbb{Z}$ tq.

$$s - (SHA(x) + dr)k_E^{-1} = jq$$

$$ss^{-1} - \frac{s^{-1}(SHA(x) + dr)}{k_E} = (s^{-1}j)q$$

$$k_E - s^{-1}(SHA(x) + dr) = (s^{-1}jk_E)q$$

que es equivalente a :

$$k_E \equiv s^{-1}SHA(x) + ds^{-1}r \bmod q$$

El lado derecho se puede expresar en terminos de valores auxiliares u_1 u_2:

$$k_E \equiv u_1 + du_2 \bmod q.$$

Demostración

Podemos plantear α a cada lado de la ecuación si reducimos modulo p:

$$\alpha^{k_E} \bmod p \equiv \alpha^{u_1+du_2} \bmod p.$$

Como el valor de la clave publica β fue calculada como $\beta \equiv \alpha^d \bmod p$ podemos escribir:

$$\alpha^{k_E} \bmod p \equiv \alpha^{u_1}\beta^{u_2} \bmod p.$$

Podemos reducir ambos lados de la ecuación modulo q:

$$(\alpha^{K_E} \bmod p) \bmod q \equiv (\alpha^{u_1}\beta^{u_2} \bmod p) \bmod q.$$

Dado que r fue construido como $r \equiv (\alpha^{k_E} \bmod p) \bmod q$ y $v \equiv (\alpha^{u_1}\beta^{u_2} \bmod p) \bmod q$ esta expresión es identica a la condición para verificar la firma:

$$r \equiv v \bmod q.$$

The Elliptic Curve Digital Signature Algorithm (ECDSA)

- Como ya sabemos las curvas elípticas tienen varias ventajas sobre RSA.
- En especial, en la ausencia de ataques potentes contra criptosistemas basados en curva elíptica (ECC).
- Longitudes de bits en el rango de 160 a 256-bits se pueden elegir, y que proporcionan una seguridad equivalente a 1.024 a 3.072 bits que utiliza el RSA o esquemas de DL.
- La longitud de bits más corta de ECC a menudo resulta en un menor tiempo de procesamiento y en las firmas más cortos.
- Dado esto, el Curve Digital Signature Algorithm Elliptic (ECDSA) se estandarizó en los EE.UU. por el American National Standards Institute (ANSI) en el 1998.

The ECDSA Algorithm

- Los pasos en el estándar ECDSA son equivalentes y estan relacionadas con el esquema DSA.
- Sin embargo, el problema del logaritmo discreto se construye en el grupo de puntos que satisfacen la curva elíptica. Por lo tanto, la operación aritmética para calcular el ECDSA es diferente de la utilizada para DSA.
- El estandar ECDSA se define para las curvas elípticas sobre cuerpos primos, es decir, $\mathbb{Z}_p$ o cuerpos de Galois $GF(2^m)$.

Key Generation

La clave para ECDSA son calculadas como:

Key Generation for ECDSA

1. Use an elliptic curve E with
   - modulus p
   - coefficients a and b
   - a point A which generates a cyclic group of prime order q
2. Choose a random integer d with $0 < d < q$.
3. Compute $B = dA$.

The keys are now:
$k_{pub} = (p, a, b, q, A, B)$
$k_{pr} = (d)$

Figure: Key Generation for ECDSA

- Observe que hemos creado un problema del logaritmo discreto donde el entero d es la clave privada y el resultado de la multiplicación escalar (dA), que es el punto B, es la clave pública.
- Al igual que el DSA, el grupo cíclico que tiene un orden q debe tener un tamaño de al menos 160-bits o más según los diferentes niveles de seguridad.

ECDSA Signature Generation

- Al igual que el DSA, una firma del ECDSA consiste en un par de enteros (r, s).
- Cada valor tiene la misma longitud en bits que q, lo que implica que las firmas son mas compactas.
- Utilizando la clave pública y privada, la firma de un mensaje x se calcula como:

ECDSA Signature Generation

1. Choose an integer as random ephemeral key k_E with $0 < k_E < q$.
2. Compute $R = k_E A$.
3. Let $r = x_R$.
4. Compute $s \equiv (h(x) + d \cdot r) {k_E}^{-1} \bmod q$.

Figure: ECDSA Signature Generation

- En el paso 3 se asigna la coordenada x del punto R a la variable r ($r = x_R$).
- El mensaje x tiene que ser hash utilizando la función h con el fin de calcular s.
- La longitud de salida de la función hash debe ser al menos tan largo como q.
- Las funciones de hash las estudiaremos más adelante en el curso.
- Por el momento la función hash comprime x y calcula una huella digital que puede ser visto como un representante de x.

ECDSA Signature Verification

- El proceso de verificación de la firma es la siguiente:

ECDSA Signature Verification

1. Compute auxiliary value $w \equiv s^{-1} \bmod q$.
2. Compute auxiliary value $u_1 \equiv w \cdot h(x) \bmod q$.
3. Compute auxiliary value $u_2 \equiv w \cdot r \bmod q$.
4. Compute $P = u_1 A + u_2 B$.
5. The verification $ver_{k_{pub}}(x,(r,s))$ follows from:

$$x_P \begin{cases} \equiv r \bmod q \Longrightarrow \text{valid signature} \\ \not\equiv r \bmod q \Longrightarrow \text{invalid signature} \end{cases}$$

Figure: ECDSA Signature Verification

- El verificador acepta una firma (r, s) sólo si el x_P tiene el mismo valor que el parámetro de la firma r modulo q.
- De lo contrario, la firma debe ser considerado inválida.

The ECDSA Algorithm

Demostración

Se demuestra que una firma (r, s) satisface la condición de verificación $r \equiv x_P mod q$.
Comenzaremos con el parámetro firma s:

$$s \equiv (h(x) + dr)k_E^{-1} \bmod q$$

que es equivalente a:

$$k_E \equiv s^{-1}h(x) + ds^{-1}r \bmod q.$$

El lado derecho se puede expresar en terminos de las valores auxiliares u_1 y u_2:

$$k_E \equiv u_1 + du_2 \bmod q.$$

Desde que un punto A genera un grupo ciclico de orden q, podemos multiplicar ambos lados de la ecuación por A:

$$k_E A = (u_1 + du_2)A$$

Como el grupo es asociativo, tenemos:

$$k_E A = u_1 A + du_2 A$$

y

$$k_E A = u_1 A + u_2 B \qquad \text{dado que} \quad B = dA.$$

- Lo que hemos demostrado hasta ahora es que la expresión

$$u_1 A + u_2 B = k_E A$$

- Si el firma, la clave correcta y el mensaje se han utilizado de manera correcta (no se ha alterado).
- Pero esto es exactamente la condición de que comprobamos en el proceso de verificación mediante la comparación de las coordenadas x de

$$P = u_1 A + u_2 B \quad \text{y} \quad R = k_E A.$$

ya que $r = x_R$ y x_P es la coordenada x del punto $P = (x, y)$